arcadian

VOLUME I August 18, 1979 Number 9

PROGRAMMABLE KEYBOARD STATUS A meeting of the FCC is scheduled around Sept.15, at which time the TI petition will be discussed. Bally currently feels that regardless of the decision, they will not be able to have a keyboard in production by the end of the year.

SURVEY The subscriber survey has resulted in a fair turnout of responses from those interested in a keyboard/memory addition, and a number of useful suggestions as well. What is evolving now is a unit that could have 16K of RAM that would accept a cassette input of the operating program, be it BASIC, COBOL, FORTRAN, or whatever(your choice) which would be loaded in about 2 minutes at 1200 baud (while the picture tube was warming up), plus an additional 8 or 16 K of onboard RAM for the user's programs. There would be space to add more RAM by chip insertion(especially the 8K version), plus connectors to allow outside memory addition. Serial and parallel ports would be available for the addition of other peripherals. Those who responded to the survey will be kept up to date.

BLACK BOX GAME enclosed is a sort of Battleship game where the computer hides some "atoms" in a grid and you have to locate them. Use the diagram for clues.

TUTORIAL ON SOUND adds more material from Chuck Thomka to last issue's discourse

SLOT MACHINE CORRECTION requires the addition of a comma to the very end of line 1515 to stop the scrolling.

BIG LETTERS continue to interest subscribers. Dennis Sprague modified the p.45 program to write double size letters on command - the poke-ing is done automatically. Refer to the program on p. 45 and retain lines 9 thru 60, and line 120. Replace the rest with:

```
65 A = 2Ø19Ø
70 K = KP
80 IF K = 13 GOTO 1ØØ
90 %(A) = K; A = A+1; GOTO 7Ø
100 %(A) = Ø
105 CLEAR
110 CALL (B); GOTO 65
```

Dennis writes " 65 starts the display area, 100 shuts ofthe display if a zero is encountered. The ASCII values of K get poked into the display area 8 bits at a time."

With the above, enter and RUN the program. The just key in whatever letter, number,character that you wish to see, punch GO, and there it is, twice as big as life.

AMERICAN CONCERT FREQUENCIES chart has bemn prepared by Robert Hood, along with the closest Bally frequency: (all in Hertz)

Note	Standard	Bally	Note	Standard	Bally
C	261.7	262	G	392	392
C#	277.3	277	G#	415.2	415
D	293.7	294	A	440	440
Eb	311.1	311	Bb	466.1	466
E	329.7	330	B	493.9	494
F	349.2	349	C	523.3	524
F#	370.1	370			

In addition, Bob has furnished a program based on the equations of p.64 to solve for frequencies or tone register values, and this is found on p.70.

BANGMAN CORRECTION COMMENT by Ernie Sams indicates that perhaps Rory Wahl has a defective logic chip if Rory's correction in the last issue is needed to make the program work. Ernie writes:
"Rory suggests that the line should read:

```
2000 E=E+1; IF E=9 GOTO 9ØØØ; IF Q≠1 GOSUB 96ØØ+(Ex1Ø)
```

Q is a flag that is set to either Ø or 1. If it is set to Ø it sends the program to the man drawing routine at 96ØØ, 961Ø,962Ø, etc.
E is a counter that is to be incremented ONLY if the guess is wrong. It is NOT to be incremented if the guess is correct or if the letter has been previously used. So the portion of the program line, E=E+1, must follow the IF statement. Now, if the 'IF' portion of the IF E=9 GOTO 9ØØØ statement is not satisfied the program defaults to the next numbered program line. The way Rory proposes, E would be incremented each time a guess is made, right or wrong. The program would never reach the man drawing routine statement because it can't go past the first IF statement until E=9 at which time the program goes to 9ØØØ, draws the gun and shoots the man that never gets drawn on the screen.
So line 2000 MUST remain exactly as was originally written or the program will not work as intended:

```
2000 IF Q≠1 GOSUB 96ØØ+(Ex1Ø); E=E+1; IF E=9 GOTO 9ØØØ
```

I have included all of Ernie's discourse as I felt that it would be of interest as a tutorial in why things are done in a certain way.

BOB HOOD's program to convert frequecies to register values and vica versa:

```
3   :RETURN
4   .FREQUENCIES
5   .ROBERT HOOD
6   .AUGUST 1979
8   NT=Ø
10  CLEAR;PRINT"BALLY
    TONE FREQUENCIES
20  PRINT"COMPUTES FREQU
    ENCY OF TONE
30  PRINT"REGISTER A B O
    R C IF VALUE OF
40  PRINT"MASTER & TONE
    REGISTERS
50  PRINT"ARE KNOWN. ALS
    O COMPUTES
60  PRINT"SETTINGS OF TO
    NE REGISTER
70  PRINT"FOR A DESIRED
    FREQUENCY
80  PRINT"IF MASTER RFGI
    STER VALUE
90  PRINT"IS KNOWN
100 PRINT"FOR FREQUENCY
    CALC INPUT 1
110 INPUT"FOR SETTING TO
    NE INPUT 2 ?"A
120 IF A=1 GOTO 15Ø
130 IF A=2 GOTO 3ØØ
140 GOTO 1ØØ
150 CLEAR;INPUT"MASTER C
    OUNTER VALUE ?"M
160 INPUT"TONE COUNTER
    VALUE ?"T
170 F=1ØØØØ÷((M+1)x112)
    ;H=RMx1Ø÷((M+1)x112)
172 G=1ØØØØ÷(T+1);I=RMx
    1Ø÷(T+1)
174 F=FxG+GxH÷9+FxI÷9+
    HxI÷9
190 PRINT;PRINT"FREQUENC
    Y IS",F,"HERTZ"
200 INPUT"INPUT 1 TO CON
    TINUE CALC."Z
210 IF Z=1CLEAR;GOTO 1ØØ
220 STOP
300 CLEAR;INPUT"INPUT DE
    SIRED FREQUENCY?"F
310 R=1ØØØØ÷F;V=RMx1Ø÷F
320 S=89
330 PRINT;INPUT"SET MAST
    ER COUNTER VALUE?"M
340 S=RxS÷(M+1)+VxS÷(M
    x9)
350 PRINT;PRINT"FOR FREQ
    UENCY OF",F
360 PRINT"MASTER COUNTER
    ",M
370 PRINT"TONE VALUE IS
    ",S
380 PRINT;PRINT"INPUT 1
    TO CONTINUE
390 INPUT"INPUT 2 TO STO
    P ?"J
400 IF J=1 CLEAR;GOTO1ØØ
410 STOP
```

TUTORIAL - SOUND SYNTHESIZER, Part 2 by Chuck Thomka

Whenever RESET is pushed, the &(16) to &(23) registers are set to fixed values.(This sort of thing is called DEFAULT) This also happens at POWER TURN ON. &(16) is set to 71, and &(17) through &(23) are set to zero.

Since pushing most keys on the keypad will generate a sound, one of the voices must be used. This means that since most keys have unique tones when pushed, they must be loading unique values into one or more of the registers. The voice used is the 'A' counter. Each key, when pushed, puts a value into the &(17) register that the 'A' counter will count up to. It will also put value 15 into the &(22) register, that will adjust the 'A' volume to its maximum so that the resultant frequency can be heard. At the end of the time of outputting the tone, the &(17) and &(22) are both put back to zero.

At anytime that the computer is stopped, the &(16) register will be set to 71, and &(17) and &(22) will be set to zero. This may affect some results of sound effects in programs where you want those registers to be left at some other values. All the other registers will be as they were last adjusted to, so remember this if you still have a tone or noise remaining after the computer has stopped.

Later in this article is a table of all the sound generating keys, their &(17) values, the resultant frequencies, and any special notes about them. (p.73)

The keys that do not generate sounds are ÷, x, +, and -. These keys will modify the sounds created by the other keys if the modifying keys are used just prior to the normal sound keys.

The divide key (÷) will make the sound one octave lower in frequency than normal. This is done by temporarily making the master counter &(16) count twice as far. So while &(16) is normally at 71, for this one note it will be set to 143. As soon as the note has finished, &(16) will again return to 71 unless the next note is also preceeded by a ÷ .

The multiply key (x) will make the sound one octave higher in frequency. This is done by making &(16) equal 35 for the time the concerned note is sounding, at the end of which the &(16) will again return to 71.

The plus (+) and minus (-) keys are only used in conjunction with the numbers 1 through 7. This was arranged so that the plus and minus sign would be meaningful in playing musical sharps or flats in the Bally-mentioned 3 octave musical scale.

Another thing to mention is the "Note Timer" or NT. For each number of NT the notes played will be approximately 17 milliseconds long. An NT=Ø results in no sound, while the maximum value of NT=255 results in about a 4.335 second note. (0.017 x 255 = 4.335)

TV 110
MC 5

The Ø is used to extend the duration of a played note by taking the note timer and increasing it an additional NT quantity for each Ø following the note to be heard. For example, say we are to play a note while NT=10, and that this note is followed by 3 zeros, the resultant NT will be 40. After playing that modified NT, the NT will again return to normal (10 in this example) until called upon again.

A funny thing about this method of extending the duration of a played note is that you still cannot play any note longer than 4.335 seconds. This is because if you had a note timer extended by way of using zeros after a printed character, and it would result in an NT>255, the final result would probably be less than 255. To explain what I mean, you have to know about binary numbers and that the NT register is only 8 bits wide. If, for example, we had an NT of 50 and that some program that we are running is to print a character followed by 5 zeros, we would expect a temporary NT result of 300 (1+5=6, 6x50=300) but an 8 bit register's maximum bit count is only 255 while a binary conversion of decimal 300 requires 9 bits (1 0010 1100). The result is that only the least 8 bits (0010 1100) will be loaded into the NT register, so NT will temporarily be 44. This you see is a lot shorter than we had at first expected and even shorter than the normal NT of 50.

TUTORIAL-SUBROUTINES

If you have a process that you want to have repeated a number of times, it is convenient and memory-saving to use the technique called SUBROUTINE, which requires the commands GOSUB and RETURN. I recently received a short program from Bret Dabel and Vince Garzoli that has this situation, and I thought that it might be of interest to all to show how a program can be modified this way. The program as it arrived is:

```
10 A=RND(32000)
20 INPUT "PLAYER #1 GUESS:" B
30 IF A=B PRINT B, "IS RIGHT"
40 IF A>B PRINT "MORE"
50 IF A<B PRINT "LESS"
60 INPUT "PLAYER #2 GUESS:" B
70 IF A=B PRINT B, "IS RIGHT"
80 IF A>B PRINT "MORE"
90 IF A<B PRINT "LESS"
100 INPUT "PLAYER #3 GUESS:" B
110 IF A=B PRINT B, "IS RIGHT"
120 IF A>B PRINT "MORE"
130 IF A<B PRINT "LESS"
140 INPUT "PLAYER #4 GUESS:" B
150 IF A=B PRINT B, "IS RIGHT"
160 IF A>B PRINT "MORE"
170 IF A<B PRINT "LESS"
180 IF A=B GOTO 10
190 GOTO 20
```

To utilize the SUBROUTINE command, we make the process to be repeated into a set of generalized statements and end them with the RETURN command. Then whenever you wish to perform the process, you direct the machine to the proper location with the GOSUB command, and when the machine does its job, it reads RETURN which tells it to go back to where it left the main program and pick up the next line number. This last statement is quite important.

As an example, lets review the Guessing Game program. We see that the A and B comparisons occur four times and so we can make a subroutine of them, giving them a set of line numbers away from the main program, as:

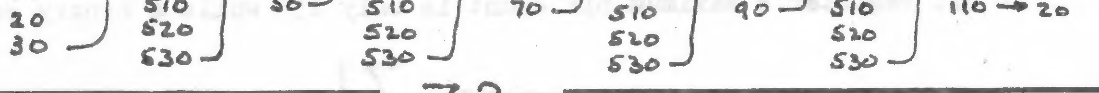
```
500 IF A = B PRINT B,"IS RIGHT"
510 IF A>B PRINT "MORE"
520 IF A<B PRINT "LESS"
530 RETURN
```

The program then reads:

```
10 A=RND(32000)
20 INPUT "PLAYER #1 GUESS:" B
30 GOSUB 500
40 INPUT "PLAYER #2 GUESS:" B
50 GOSUB 500
60 INPUT "PLAYER #3 GUESS:" B
70 GOSUB 500
80 INPUT "PLAYER #4 GUESS:" B
90 GOSUB 500
100 IF A=B GOTO 10
110 GOTO 20
500 IF A=B PRINT B, "IS RIGHT"
510 IF A>B PRINT "MORE"
520 IF A<B PRINT "LESS"
530 RETURN
```

If by chance you have written the subroutine at lines 500 - 530 but later you have added so much program that 500-530 will be buried in the program length, you will have a problem. As the computer completes line 490, it will search for the last value of A and B and perform the comparisons asked for in lines 500 through 520 (would you want that, then?) but it will HOW? when it gets to 530 because it is not in a subroutine mode and has no place to return to. To avoid this, you jump around the subroutine, in our case with a 490 GOTO 540. Since this is a legitimate operation, it means therefore that the subroutine could actually be placed anywhere within the program, and a suitable jump statement added.

Nesting of subroutines is possible. By this we mean that once you have gotten into the subroutine loop, you could have another subroutine called. The machine would perform the second subroutine and RETURN to the next line number of the first subroutine, and on its completion, go back to the main program. I believe that four such 'nests' are possible in the Bally BASIC, but one has to be very careful that each subroutine loop is completed - there cannot be any open loops. Diagramatically, the guessing game looks like this:



&(17) VALUE	CHARACTER(S)	RESULTANT FREQUENCY NORMAL &(16)= 71	÷ PREFIX &(16)=143	x PREFIX &(16)=35
225	!	54.63 HZ	27.32 HZ	109.26 HZ
212	"	57.97	28.98	115.93
200	#	61.43	30.71	122.85
189	$	64.98	32.49	129.97
178	%	68.98	34.49	137.95
168	&	73.06	36.53	146.12
159	' (APOST.)	77.17	38.58	154.34
150	(	81.77	40.88	163.53
141	)	86.95	43.47	173.90
133	*	92.14	46.07	184.28
119	, (COMMA)	102.89	51.45	205.78
106	. (PERIOD)	115.39	57.70	230.78
100	/ Y -1	122.25	61.12	244.49
94	Z 1	129.97	64.98	259.93
89	[+1 -2	137.19	68.59	274.37
84	\ 2	145.26	72.63	290.51
79	] +2 -3	154.34	77.17	308.67
74	↑ 3	164.62	82.31	329.25
70	← 4	173.90	86.95	347.80
66	↓ +4	184.28	92.14	368.56
62	→ 5	195.98	97.99	391.96
59	+5 -6	205.78	102.89	411.56
55	6	220.48	110.24	440.96
52	+6 -7	232.96	116.48	465.92
49	7	246.94	123.47	493.87
46	+7 8	262.70	131.35	525.40
44	9	274.37	137.19	548.75
41	:	293.97	146.99	587.94
39	;	308.67	154.34	617.34

&(17) VALUE	CHARACTER(S)	RESULTANT FREQUENCY NORMAL &(16)= 71	÷PREFIX &(16)=143	x PREFIX &(16)= 35
37	<	324.92 HZ	162.46 HZ	649.83 HZ
34	=	352.77	176.38	705.53
32	>	374.15	187.07	748.29
31	?	385.84	192.92	771.68
29	@	411.56	205.78	823.12
27	A	440.96	220.48	881.91
26	B	457.29	228.64	914.58
24	C	493.87	246.94	987.74
23	D	514.45	257.23	1028.90
21	E	561.22	280.61	1122.44
20	F	587.94	293.97	1175.89
19	G	617.34	308.67	1234.68
18	H	649.83	324.92	1299.66
17	I	685.93	342.97	1371.87
16	J	726.28	363.14	1452.57
15	K	771.68	385.84	1543.35
14	L	823.12	411.56	1646.24
13	M	881.91	440.96	1763.83
11	N	1028.90	514.45	2057.80
10	O	1122.44	561.22	2244.87
9	P	1234.68	617.34	2469.36
8	Q	1371.87	685.93	2743.73
7	R	1543.35	771.68	3086.70
6	S	1763.83	881.91	3527.66
5	T	2057.80	1028.90	4115.60
4	U	2469.36	1234.68	4938.72
3	V	3086.70	1543.35	6173.40
2	W	4115.60	2057.80	8231.20
1	X	6173.40	3086.70	12346.81

A;RETURN

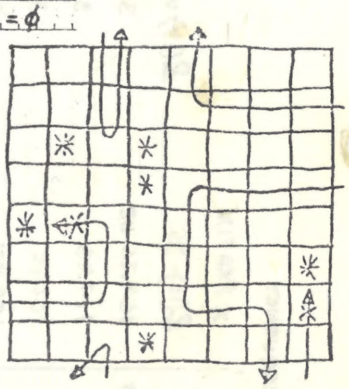
```
Line #   Statement(s)
   1   .BLACK BOX
   2   .BY B.REANY
  10   BC=1A
  15   BC=BC+16;FC=ϕ
  20   CLEAR
  25   @(1)=2
  30   @(2)=2
  35   NT=5ϕ;CX=-41
  40   PRINT" BLACK BOX
  50   CX=-47;CY=ϕ;NT=ϕ
  55   PRINT"HOW MANY ATOMS
  60   CX=-41
  65   PRINT"DO YOU WANT?
  70   CY=-32;INPUT A
 100   FOR B=1 TO A
 105   C=RND(8)+1ϕ×RND(8)
 110   IF@(C)=1 B=B-1
 115   @(C)=1
 120   NEXT B
 125   CLEAR;CY=32
 135   CX=-29;PRINT"ϕ 1 2 3 4
        5 6 7
 140   FOR B= 1ϕ TO 17
 145   CX=-47;PRINT#2,B,;CX=67;
       PRINT#2,B+2ϕ
 150   NEXT B
 180   CX=-35;PRINT"2ϕ 1 2 3 4
        5 6 7",;CY=4ϕ
 185   BOX 13,-4,97,65,3
 190   FOR B=-29TO55 STEP 12
 195   FOR C=24 TO-32 STEP-8
 200   BOX B,C,11,7,3
 205   NEXT C
 210   NEXT B
 250   FOR B=1TO 24ϕϕ
 255   IF&(22)=16 GOTO 15
 260   IF&(23)=8 GOTO 4ϕϕ
 265   IF&(22)=8 GOTO 3ϕϕ
 270   IF&(21)=8 GOTO 8ϕϕ
 275   NEXT B

Line #   Statement(s)
 280   FC=9;GOTO 25ϕ
 300   CX=-77;CY=4ϕ;INPUT C
 310   CX=-41;CY=4ϕ;INPUT R
 315   C=12×C-29;R=24-8×(R-1ϕ)
 320   BOX C,R,3,3,3
 325   CY=4ϕ;PRINT"                 ",
 330   NT=3;GOTO 25ϕ
 400   CY=4ϕ;CX=-77;INPUTR
 405   NT=ϕ;@(2)=@(2)+1
 410   B=R÷1ϕ;C=RM+1
 415   IFB=0 J=C;K=ϕ;L=ϕ;M=1
 420   IFB=1 J=ϕ;K=C;L=1;M=ϕ
 425   IFB=2 J=C;K=9;L=ϕ;M=-1
 430   IFB=3 J=9;K=C;L=-1;M=ϕ
 450   CX=-77;CY=4ϕ;D=1ϕ×J+K
 460   IF@(D)=1 PRINT"ABSORBED";
       GOTO 325
 465   IF L=ϕ S=D+1ϕ+M;T=S-2ϕ
 470   IF M=ϕ S=D+1+1ϕ×L;T=S-2
 471   IF T<1 T=1
 475   IF@(S)=1 IF@(T)=1 L=-L;
        M=-M; GOTO 5ϕϕ
 480   IF L=ϕ IF@(S)=1 L=-1;M=ϕ;
        GOTO 5ϕϕ
 485   IF L=ϕ IF@(T)=1 L=1;
        M=ϕ;GOTO 5ϕϕ
 490   IF M=ϕ IF@(S)=1 M=~~-M~~-1;
        L=ϕ;GOTO 5ϕϕ
 495   IF M=ϕ IF@(T)=1 M=1;L=ϕ
 500   J=J+L; K=K+M
 505   IF J<1 GOTO 55ϕ
 510   IF J>8 GOTO 55ϕ
 515   IF K<1 GOTO 55ϕ
 520   IF K>8 GOTO 55ϕ
 525   GOTO 45ϕ
 550   K=32-8×K;J=12×J-41
 560   NT=5ϕ
 565   FOR B=1 TO 5ϕ
 570   BOX J,K,11,7,3
 575   NEXT B

Line #   Statement(s)
 580   NT=3;GOTO 325
 800   FOR B=11 TO 88
 805   C=B÷1ϕ-1;D=RM-1
 820   C=-29+C×12;D=24-D×8
 830   IF@(B)=1 BOX C,D,7,5,
        3; @(B)=ϕ
 840   IF PX(C,D)=1 @(1)=@(1)+1
 850   NEXT B
 855   NT=5ϕ;CY=4ϕ;CX=-35
 865   PRINT "F I N I S H
 870   NT=3; CLEAR
 875   IF @(1)>1ϕ GOTO 9ϕϕ
 880   PRINT"EUREKA!
 885   CY=ϕ;PRINT"YOU HAD ",
        #2,A,"ATOMS
 890   PRINT"YOU USED ",#1,
        @(2)-2,"RAYS
 895   GOTO 25ϕ
 900   PRINT" SORRY
 910   GOTO 25ϕ
```

LINE 145 put ; between B and ;
LINE 490 change -M to -1

See Corrections
Vol 1 page 77



BLACK BOX RULES:

An 8 by 8 grid has a predetermined number of atoms hidden, one per square, under the grid squares. Berthold rays will be generated after you select a ray entry point after pressing the "1" key. Rays travel in straight lines perpendicular to the grid edge, starting from the ray entry point, until they are absorbed or exit from the grid. They obey the following rules;

1. A ray entering the grid on either side of an atom on the edge of the grid is deflected backward and away from the edge atom.
2. A ray aimed between two atoms with an open square between them is reflected back upon its' path.
3. A ray coming within one square diagonally of an atom is deflected away 90 degrees from that atom.
4. A ray colliding with an atom will be absorbed, and its' absorption will be signalled upon the screen
5. A ray emerging from the grid will signal its' exit point.

The "2" key will ask for a col (0 to 7) and row (10 to 17) and will either mark or unmark the grid position at their intersection where you suspect an atom is located. The "3" key will display the grid points where the atoms were located, those which you marked as having atoms, and will grade your guesses, and clear the grid for the next game. The zero key will restart the game, but if the grid was not cleared with the "3" key, the preceding grid atoms will not be cleared.

After this program is loaded, the direct executed "PRINT SZ" command must print at least 200, or the program strings will be insufficient to execute. For this reason, closing quote marks on literals, as well as several obvious input edits, have been deleted.

This program is unconditionally guaranteed by the author to be smack up against your core limitation, or double your core dumps back.

Thank You
B. Reany
1106 E. Julia Dr.
Perry, Fl 32347

POKE-ING PROGRAM allows you to load machine instructions into the @ string, which means that you can call several machine language subroutines from inside the BASIC. Developed by George Breadon, the program follows along with some data to be inserted that will call up our old buddy, ARCADIAN (ref.p.45)

```
  5 NT = Ø
 10 A = 2Ø18Ø; B=A; For K=Ø TO 13 }       {INPUT MACHINE INSTRUCTIONS
 20 INPUT @(K); NEXT K                    {INTO @ STRING
 30 FOR K = Ø TO 13; CLEAR }
 40 CY = Ø;PRINT K,@(K)                   {EDIT ROUTINE- HIT "STEP"
 50 D = KP; IF D=31 GOTO 8Ø                KEY (D=57) TO STEP THRU
 60 IF D=57 GOTO 9Ø                        MACHINE INSTRUCTIONS.
 70 GOTO 5Ø                       ------   HIT "ERASE" KEY (D=31) TO
 80 INPUT "CHANGE=",L;@(K)=L               CHANGE MACHINE INSTR.
 90 NEXT K
100 A=B; FOR K= Ø TO 13 } ------------    POKE @STRING INTO MEMORY
110 %(A)=@(K); A=A+2; NEXT K
120 IF &(20) = 8 GOTO 3Ø                  {HIT "GOTO" KEY TO BRANCH
130 C=2Ø18Ø; GOSUB 16Ø } ---------------  {BACK TO EDIT ROUTINE AT
140 C=2Ø19Ø; GOSUB 16Ø }                  {ANY TIME
150 GOTO 12Ø
160 CLEAR; CALL (C); RETURN ------------  {INITIALIZE STARTING ADDRESS
                                          {FOR SUBROUTINE 52
                                           CALL SUBROUTINE 52
```

DATA to be inserted: This is all in machine level code.

@(Ø) =	-43	@(7) =	27672
1	12341	8	20200
2	19480	9	-13871
3	3164 Or 3159	10	21057
4	-13871	11	16707
5	-43	12	18756
6	53	13	20033

@ 0 thru 4 go into 20180 while @ 5 thru 13 go into 20190, two at a time

SOFTWARE PRODUCERS are invited to contact VIDEO CONCEPTS at 625 W. 53 Ave, Anchorage Alaska, 99502, for distribution of their products thru the store up in the cold country.

RETURNED BALLY UNITS are available from V. Jupe, Star Route Box 60, Carlotta, CA, 95528 These are working, and at less than $200. Also some games, write.

ADS start here this time:

SELL ARCADE with 4 controls, BASIC cassette and interface. BLACKJACK,BASEBALL, RED BARON, CROSSWORDS Interface has jack for printer. Reasonable Offer to Bob Schwind 12311 W. Silver Spring Dr. Milwaukee, WI 53225 (414) 367-4804

1342 1703 1622 1713 1022 1631
164 165 171 [illegible] 164
1711 1722 89

ADS

SELL ARCADE complete, includes PANZER ATTACK, CLOWNS, ESCAPE, FOOTBALL, BASEBALL, BINGO MATH, LETTER MATCH, PLACKJACK, ETC., BASIC and CASSETTE INTERFACE. Total original list price 560. First certified check for 400, or best offer. B.PERLSON 6400 N.EIM TREE RD. MILWAUKEE WI, 53217 414-352-1331

Quality games on C-10 cassettes: STARBLASTER (2 player spacewar) and HAMMURABI (you control ancient Sumeria) at 7.each, both for 12. Dan. Pierce 229 Orville St Apt 1 Fairborn OH 45324

LISTING for the game SUB SEARCH, a one-player item, at $1.25 Marc Gladstein 1213½ S. ALFRED ST Los Angeles CA 90035 (213)658-5804

Available through Sebree's Computing 456 Granite Ave., Monrovia CA 91016- Games:3.95-UFO BATTLE, HIT THE PEDESTRIAN, SUBMARINE MINEFIELD; 2.95-MUNCH!;5.50-DOWN THE TRENCH;$8,95-**SUPER WUMPUS**;$2.50-MATH ROUTINES (calculates Sine, Cosine, Arctangent, & Square Root!!). All programs with one page of documentation/instructions. Send for descriptions. Timothy Hays.

A note from W&W Software that they have another cassette ready.

SELL Bally ARCADE BPA 1100 with BASIC, FOOTBALL, BASEBALL, 4 other cassettes, tape interface $275. Geo. Evanoff, 10028 N.E. 28th Place, Bellevue WA 98004 (206)-827-2918

One player game called SUBSEARCH, 1.25 for listing, only. Marc Gladstein 1213½ S. Alfred St. Los Angeles CA 90035 (213) 658-5804

REVIEW of programs has been suggested by some subscribers, who are concerned about purchasing a 'pig in a poke'. IF someone else is willing to do a critical review of a program that some advertiser is also willing to submit, I will get the two parties together and accept the review for publication. The opinions will be the reviewers, not mine.

ARCADIAN
Robert Fabris, stamp licker
3626 Morrie Dr.
San José, CA 95127

SAN JOSE, CA 951
PM
26 AUG
1979

USA 15c
THE LAND OF THE FREE
THE HOME OF THE BRAVE

FIRST CLASS

94550HUS0635L
R. HAUSER
635 LOS ALAMOS AVENUE
LIVERMORE, CA 94550